INPI
INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE

BREVET D'INVENTION

CERTIFICAT D'UTILITÉ - CERTIFICAT D'ADDITION

COPIE OFFICIELLE

Le Directeur général de l'Institut national de la propriété industrielle certifie que le document ci-annexé est la copie certifiée conforme d'une demande de titre de propriété industrielle déposée à l'Institut.

Fait à Paris, le 05 FEV. 2004

Pour le Directeur général de l'Institut
national de la propriété industrielle
Le Chef du Département des brevets

Martine PLANCHE

UMENT DE PRIORITÉ

SENTÉ OU TRANSMIS
NFORMÉMENT À LA
ÈGLE 17.1.a) OU b)

INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE
SIEGE
26 bis, rue de Saint Petersbourg
75800 PARIS cedex 08
Téléphone : 33 (0)1 53 04 53 04
Télécopie : 33 (0)1 53 04 45 23
www.inpi.fr

ETABLISSEMENT PUBLIC NATIONAL CREE PAR LA LOI N° 51-444 DU 19 AVRIL 1951

INPI
INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE

26 bis, rue de Saint Pétersbourg
75800 Paris Cedex 08
Téléphone : 33 (1) 53 04 53 04 Télécopie : 33 (1) 42 94 86 54

BREVET D'INVENTION
CERTIFICAT D'UTILITÉ

Code de la propriété intellectuelle - Livre VI

cerfa N° 11354*03

REQUÊTE EN DÉLIVRANCE
page 1/3

BR1

Cet imprimé est à remplir lisiblement à l'encre noire DB 540 @ B / 210502

Réservé à l'INPI

REMISE DES PIÈCES
DATE 24 DEC 2002
LIEU 75 INPI PARIS

N° D'ENREGISTREMENT NATIONAL ATTRIBUÉ PAR L'INPI 0216652

DATE DE DÉPÔT ATTRIBUÉE PAR L'INPI 24 DEC. 2002

Vos références pour ce dossier *(facultatif)* 10B610 12FR001/LCH

1 NOM ET ADRESSE DU DEMANDEUR OU DU MANDATAIRE À QUI LA CORRESPONDANCE DOIT ÊTRE ADRESSÉE

BOUJU DERAMBURE BUGNION
52 rue de Monceau
75008 PARIS

Confirmation d'un dépôt par télécopie ☐ N° attribué par l'INPI à la télécopie

2 NATURE DE LA DEMANDE — Cochez l'une des 4 cases suivantes

Demande de brevet	☒	
Demande de certificat d'utilité	☐	
Demande divisionnaire	☐	
Demande de brevet initiale	N°	Date
ou demande de certificat d'utilité initiale	N°	Date
Transformation d'une demande de brevet européen *Demande de brevet initiale*	☐ N°	Date

3 TITRE DE L'INVENTION (200 caractères ou espaces maximum)

Procédé et dispositif de vérification de l'intégrité d'une application logicielle sans clé de chiffrement/déchiffrement

4 DÉCLARATION DE PRIORITÉ OU REQUÊTE DU BÉNÉFICE DE LA DATE DE DÉPÔT D'UNE DEMANDE ANTÉRIEURE FRANÇAISE

Pays ou organisation
Date N°
Pays ou organisation
Date N°
Pays ou organisation
Date N°

☐ **S'il y a d'autres priorités, cochez la case et utilisez l'imprimé «Suite»**

5 DEMANDEUR (Cochez l'une des 2 cases) ☐ Personne morale ☒ Personne physique

Nom ou dénomination sociale		BANGUI
Prénoms		François
Forme juridique		
N° SIREN		
Code APE-NAF		
Domicile ou siège	Rue	69 rue Dunois
	Code postal et ville	7 5 0 1 3 PARIS
	Pays	FRANCE
Nationalité		Française
N° de téléphone *(facultatif)*		N° de télécopie *(facultatif)*
Adresse électronique *(facultatif)*		

☐ **S'il y a plus d'un demandeur, cochez la case et utilisez l'imprimé «Suite»**

Remplir impérativement la 2ème page

INPI
INSTITUT NATIONAL DE LA PROPRIÉTÉ INDUSTRIELLE

BREVET D'INVENTION
CERTIFICAT D'UTILITÉ

REQUÊTE EN DÉLIVRANCE
page 2/3

BR2

Réservé à l'INPI

REMISE DES PIÈCES
DATE 24 DEC 2002
LIEU 75 INPI PARIS
N° D'ENREGISTREMENT NATIONAL ATTRIBUÉ PAR L'INPI 0216652

DB 540 W / 210502

6 MANDATAIRE *(s'il y a lieu)*		
Nom		
Prénom		
Cabinet ou Société		BOUJU DERAMBURE BUGNINO
N °de pouvoir permanent et/ou de lien contractuel		
Adresse	Rue	52 rue de Monceau
	Code postal et ville	7 5 0 0 8 PARIS
	Pays	FRANCE
N° de téléphone *(facultatif)*		01 45 61 51 00
N° de télécopie *(facultatif)*		01 45 61 96 30
Adresse électronique *(facultatif)*		mail@bdsa.com
7 INVENTEUR (S)		**Les inventeurs sont nécessairement des personnes physiques**
Les demandeurs et les inventeurs sont les mêmes personnes		☒ Oui ☐ Non : **Dans ce cas remplir le formulaire de Désignation d'inventeur(s)**
8 RAPPORT DE RECHERCHE		**Uniquement pour une demande de brevet (y compris division et transformation)**
Établissement immédiat ou établissement différé		☒ ☐
Paiement échelonné de la redevance *(en deux versements)*		**Uniquement pour les personnes physiques effectuant elles-mêmes leur propre dépôt** ☐ Oui ☐ Non
9 RÉDUCTION DU TAUX DES REDEVANCES		**Uniquement pour les personnes physiques** ☐ Requise pour la première fois pour cette invention *(joindre un avis de non-imposition)* ☐ Obtenue antérieurement à ce dépôt pour cette invention *(joindre une copie de la décision d'admission à l'assistance gratuite ou indiquer sa référence)* : **AG**
10 SÉQUENCES DE NUCLEOTIDES ET/OU D'ACIDES AMINÉS		☐ Cochez la case si la description contient une liste de séquences
Le support électronique de données est joint		☐
La déclaration de conformité de la liste de séquences sur support papier avec le support électronique de données est jointe		☐
Si vous avez utilisé l'imprimé «Suite», indiquez le nombre de pages jointes		

11 SIGNATURE DU DEMANDEUR OU DU MANDATAIRE
(Nom et qualité du signataire)

Le Mandataire
Olivier NICOLLE
92 - 3040

VISA DE LA PRÉFECTURE OU DE L'INPI

La loi n°78-17 du 6 janvier 1978 relative à l'informatique, aux fichiers et aux libertés s'applique aux réponses faites à ce formulaire. Elle garantit un droit d'accès et de rectification pour les données vous concernant auprès de l'INPI.

INPI
INSTITUT NATIONAL DE LA PROPRIETE INDUSTRIELLE

26 bis, rue de Saint Pétersbourg
75800 Paris Cedex 08
Téléphone : 33 (1) 53 04 53 04 Télécopie : 33 (1) 42 94 86 54

BREVET D'INVENTION
CERTIFICAT D'UTILITÉ

Code de la propriété intellectuelle - Livre VI

cerfa N° 11354*03

REQUÊTE EN DÉLIVRANCE

Page suite N° 3 . . /3 . . . BR/SUITE

Réservé à l'INPI

REMISE DES PIÈCES
DATE 24 DEC 2002
LIEU 75 INPI PARIS

N° D'ENREGISTREMENT NATIONAL ATTRIBUÉ PAR L'INPI 0216652

Cet imprimé est à remplir lisiblement à l'encre noire

DB 829 ⊕ W / 010702

Vos références pour ce dossier *(facultatif)*		10B610 12FR001/LCH
4 **DÉCLARATION DE PRIORITÉ OU REQUÊTE DU BÉNÉFICE DE LA DATE DE DÉPÔT D'UNE DEMANDE ANTÉRIEURE FRANÇAISE**		Pays ou organisation Date N°
		Pays ou organisation Date N°
		Pays ou organisation Date N°
5 **DEMANDEUR** (Cochez l'une des 2 cases)		☐ Personne morale ☒ Personne physique
Nom ou dénomination sociale		GONTIER
Prénoms		William
Forme juridique		
N° SIREN		
Code APE-NAF		
Domicile ou siège	Rue	83 rue de Nantes
	Code postal et ville	77290 MITRY MORY
	Pays	FRANCE
Nationalité		Française
N° de téléphone *(facultatif)*		
N° de télécopie *(facultatif)*		
Adresse électronique *(facultatif)*		
5 **DEMANDEUR** (Cochez l'une des 2 cases)		☐ Personne morale ☐ Personne physique
Nom ou dénomination sociale		
Prénoms		
Forme juridique		
N° SIREN		
Code APE-NAF		
Domicile ou siège	Rue	
	Code postal et ville	
	Pays	
Nationalité		
N° de téléphone *(facultatif)*		
N° de télécopie *(facultatif)*		
Adresse électronique *(facultatif)*		

16 **SIGNATURE DU DEMANDEUR OU DU MANDATAIRE (Nom et qualité du signataire)**

Le Mandataire
Olivier NICOLLE
92 - 3040

VISA DE LA PRÉFECTURE OU DE L'INPI

La loi n°78-17 du 6 janvier 1978 relative à l'informatique, aux fichiers et aux libertés s'applique aux réponses faites à ce formulaire.
Elle garantit un droit d'accès et de rectification pour les données vous concernant auprès de l'INPI

La présente invention concerne la vérification de l'authenticité d'une application logicielle exécutée sur un terminal hôte sans nécessairement faire appel à des clés de chiffrement/déchiffrement.

Elle trouve une application générale dans l'authentification d'applications logicielles et plus particulièrement les applications logicielles destinées à être exécutées sur un dispositif de traitement de données, notamment un terminal hôte tel qu'un décodeur de télévision numérique, un équipement de visualisation de contenus multimédia, un micro-ordinateur, une carte à puce, un assistant personnel, une console de jeux, un téléphone mobile ou analogue.

On connaît déjà des moyens d'authentification permettant de vérifier l'authenticité ou l'intégrité d'applications logicielles embarquées et exécutées sur des terminaux hôtes. Généralement, de tels moyens d'authentification mettent en œuvre des fonctions de hachage et/ou des algorithmes cryptographiques qui utilisent des données secrètes telles que des clés privées ou secrètes de chiffrement/déchiffrement cachées dans le logiciel de vérification du terminal hôte. Le plus souvent, ces données secrètes sont protégées par des techniques d'offuscation destinées à rendre plus difficile la rétro-conception.

En pratique, de tels moyens d'authentification embarqués dans les terminaux hôtes sont tout aussi vulnérables que les applications logicielles dont ils sont sensés contrôler l'authenticité. En effet, un pirate averti peut opérer des modifications malveillantes sur ces moyens d'authentification, afin, par exemple, de récupérer les données secrètes, leurrer le système de vérification ou lui faire produire, malgré lui, les résultats attendus.

La présente invention remédie à cet inconvénient.

Elle porte sur un procédé de vérification de l'intégrité d'une application logicielle exécutable dans un terminal hôte.

Selon une définition générale de l'invention, le procédé comprend les étapes suivantes :

i) déterminer au moins une suite d'instructions de contrôle formant certificat exécutable pour ladite application logicielle,

ii) sur le terminal hôte, exécuter l'application logicielle à vérifier, recevoir le certificat exécutable ainsi déterminé lors de l'étape i), et exécuter la suite d'instructions de contrôle dudit certificat exécutable dans le contexte mémoire dudit terminal hôte,

iii) comparer le résultat ainsi obtenu par l'exécution des instructions de contrôle avec le résultat attendu d'une application authentique et,

iv) en cas de comparaison positive, continuer l'exécution de l'application logicielle à vérifier.

On entend ici par le terme « comparaison positive » le fait que toute action, opération, ou modification sur les données utilisées par l'application logicielle à vérifier ou toute action, opération ou modification sur le déroulement de l'exécution de l'application logicielle à vérifier produise un comportement de l'application logicielle à vérifier identique à celui qui est attendu par le déroulement de l'exécution de l'application authentique.

En renouvelant, à une cadence choisie, la suite d'instructions de contrôle du certificat exécutable, il est possible de mettre en œuvre un nombre important de moyens d'authentification d'une application et il devient quasi impossible de mettre au point des applications logicielles pirates qui déjouent systématiquement le processus de vérification.

Ainsi, le procédé selon l'invention permet de vérifier l'intégrité d'une application logicielle exécutée sur un terminal hôte avec un degré de sécurisation relativement satisfaisant vis à vis des pirates adeptes de la rétro-conception et cela sans faire appel à des clés de chiffrement/déchiffrement ou à des composants matériels coûteux.

Selon une réalisation, la suite d'instructions de contrôle est choisie de telle sorte que l'état du contexte mémoire d'une application logicielle authentique

après l'exécution de la suite d'instructions de contrôle est identique (sans modification) à l'état du contexte mémoire de l'application logicielle avant l'exécution de la suite d'instructions de contrôle.

Ainsi la mise en œuvre du procédé selon l'invention n'apporte pas de dysfonctionnement au niveau du déroulement de l'application logicielle à vérifier si cette dernière est authentique.

Selon une réalisation, dans laquelle le terminal hôte est équipé d'un processeur, la suite d'instructions de contrôle formant certificat exécutable est codée en langage interprétable par ledit processeur du terminal hôte.
En variante, dans laquelle le terminal hôte est équipé d'une machine virtuelle apte à émuler un processeur, la suite d'instructions de contrôle formant certificat exécutable est codée en langage interprétable par la machine virtuelle du terminal hôte.

En pratique, le contexte mémoire d'exécution d'une application logicielle est constitué, entre autre, des adresses en mémoire des symboles (fonctions, variables,...), des instructions exécutables, des données, et de l'état de la pile d'exécution de l'application. Ces valeurs sont uniques pour chaque application informatique en cours d'exécution et pour chaque type de processeur ou de machine virtuelle.

En pratique, dans l'étape i) il est prévu d'établir, dans un environnement sécurisé, une carte du contexte mémoire de l'application logicielle authentique en cours d'exécution, de déterminer, en utilisant les valeurs de cette carte mémoire, une suite d'instructions de contrôle formant certificat exécutable.

En pratique, dans l'étape ii), le dispositif d'acheminement du certificat exécutable à destination du terminal hôte est logé dans un circuit électronique de traitement physiquement séparé du terminal hôte.

En pratique, dans l'étape ii) la récupération des valeurs du contexte mémoire d'exécution se fait par lecture des valeurs aux adresses des différentes zones de mémoire du terminal hôte. Dans ces zones sont logées les instructions exécutables intrinsèques à l'application, les valeurs des variables et les valeurs des références aux fonctions de l'application à contrôler.

En pratique, à l'étape iii), le résultat obtenu par l'exécution de ladite suite d'instructions de contrôle est une signature de l'application à vérifier. Cette signature est calculée par ladite suite d'instructions de contrôle qui utilise les valeurs du contexte mémoire de l'application logicielle à vérifier en cours d'exécution. De préférence, l'application logicielle comprend des instructions permettant d'insérer et d'exécuter dans son contexte mémoire ladite suite d'instructions en substituant au moins une adresse d'exécution d'une instruction de ladite application logicielle par au moins l'adresse d'une instruction de la suite d'instructions de contrôle formant certificat exécutable.

Selon une autre réalisation, la suite d'instructions de contrôle formant certificat exécutable est transportée dans un flux de données nécessaire à l'exécution de l'application logicielle à vérifier. Pour forcer l'exécution de la suite d'instructions dudit certificat exécutable, lesdites données utiles sont préalablement protégées par une méthode de chiffrement. Le déchiffrement de ces données est correctement effectué par ladite suite d'instructions de contrôle du certificat exécutable si l'application à vérifier est une application authentique. Si le procédé de chiffrement utilise une clé, cette dernière est produite par les instructions de contrôle avec des valeurs du contexte mémoire de l'application logicielle à vérifier, valeurs formant signature de l'application logicielle à vérifier. Les opérations pour obtenir la clé de chiffrement sont codées dans la suite d'instructions du certificat exécutable.

En variante, la méthode de protection est sans clé, la suite d'opérations pour obtenir l'accessibilité des données est dans la suite d'instructions de contrôle du certificat exécutable.

En pratique, la protection des données nécessaires au fonctionnement de l'application logicielle à vérifier est entreprise dans un environnement sécurisé avant que ces données ne soient transmises. La méthode de protection, avec ou sans clé, doit être réversible.

Selon encore une autre réalisation, dans laquelle l'exécution de l'application logicielle fait appel à une carte à puce ou à tout autre circuit sécurisé pour fonctionner, la suite d'instructions de contrôle est logée dans la carte à puce (ou le circuit sécurisé) et envoyée à l'application logicielle à vérifier, l'application logicielle étant apte à récupérer et exécuter ladite suite d'instructions de contrôle ainsi envoyée avec les données dont elle a besoin pour fonctionner.

En pratique, l'accès à des données transmises par la carte à puce (ou le circuit sécurisé) doit être nécessaire à l'application logicielle à vérifier pour que celle-ci se comporte de façon identique à une application authentique.

Selon une autre réalisation, à la suite d'une vérification négative de l'intégrité de l'application logicielle à vérifier, le certificat exécutable exécute des instructions faisant appel à des fonctions appartenant à une autre application.

La présente invention a également pour objet un dispositif de vérification de l'intégrité d'une application logicielle pour la mise en œuvre du précédé selon l'invention.

Selon une autre caractéristique importante de l'invention, le dispositif de vérification comprend des moyens de traitement aptes à déterminer au moins une suite d'instructions de contrôle formant certificat exécutable pour l'application logicielle, exécutable par ledit terminal hôte au cours de l'exécution de l'application logicielle à vérifier, et des moyens de comparaison pour comparer le résultat de l'exécution du certificat exécutable sur le comportement de l'application logicielle à vérifier, avec le résultat attendu du comportement d'une application authentique, et des moyens de modifier l'exécution de l'application logicielle à vérifier en fonction du résultat de la comparaison.

Selon une réalisation, le dispositif de vérification comprend une carte à puce ou tout autre circuit sécurisé apte à contenir d'une part, la suite d'instructions de contrôle formant certificat exécutable et d'autre part, une application réalisant le test de vérification. Le terminal hôte est équipé d'un lecteur de carte à puce (ou d'un moyen de communication avec le circuit sécurisé) et les moyens d'exécution de l'application logicielle à vérifier sont agencés pour charger et exécuter dans son contexte mémoire la suite d'instructions formant certificat. L'application de vérification dans la carte à puce ou le circuit sécurisé est agencée de façon à modifier le déroulement normal de l'exécution de l'application logicielle à vérifier si le résultat de l'exécution de la suite d'instructions de contrôle n'est pas transmis, dans des conditions définies préalablement, à l'application de vérification dans la carte à puce ou du circuit sécurisé, ou si le résultat de la vérification s'avère négatif.

Selon une variante, le dispositif est apte à déterminer une pluralité de certificats exécutables différents les uns des autres selon une cadence et/ou condition choisie.

En pratique, le terminal hôte appartient au groupe formé par les dispositifs de traitement des données, les décodeurs de télévision numérique, les équipements de visualisation de contenus multimédias, les micro-ordinateurs, les cartes à puces, les assistants personnels, les consoles de jeux, les téléphones mobiles ou analogues.

D'autres caractéristiques et avantages de l'invention apparaîtront à la lumière de la description détaillée ci-après et des dessins dans lesquels :

- la figure 1 est une vue schématique illustrant la vérification d'une application logicielle dont les données utiles contiennent le certificat exécutable, selon l'invention, et
- la figure 2 est une vue schématique illustrant la vérification d'une application logicielle utilisant une carte à puce selon l'invention.

En pratique, les termes « intègre » et « authentique » sont ici utilisés indifféremment pour une application logicielle.

On entend ici par « certificat exécutable » une suite d'instructions de contrôle exécutables dans le contexte mémoire d'une application logicielle en cours d'exécution et dont l'exécution produit des effets tels que l'exécution d'une l'application logicielle à vérifier, si cette dernière est intègre, a un comportement identique à celui qui est attendu.

En référence à la figure 1, l'application logicielle à vérifier 1 est embarquée dans un terminal hôte (non représenté).

Par exemple, le terminal hôte appartient au groupe formé par les dispositifs de traitement des données, les décodeurs de télévision numérique, les équipements de visualisation de contenus multimédias, les micro-ordinateurs, les cartes à puces, les assistants personnels, les consoles de jeux, les téléphones mobiles ou analogues.

En pratique, l'application logicielle à vérifier 1 traite 3 des données préalablement protégées 2, c'est a dire non traitables par l'application à vérifier tant que des instructions de contrôle du certificat exécutable que l'on décrira plus en détail ci-après n'ont pas authentifiées l'application à vérifier. Une méthode pour rendre non accessible ces données consiste à les chiffrer. Toute autre méthode de protection réversible est envisageable.

Ces données protégées 2 contiennent 7 un certificat exécutable 4 renfermant une suite d'instructions de contrôle non protégées, qui sont exécutées 5 par l'application à vérifier 1.

En pratique, les instructions de contrôle du certificat exécutable 4 sont codées dans le langage du processeur du terminal hôte, encore appelé langage machine. En variante, les instructions du certificat exécutable 4 peuvent aussi

être codées dans le langage d'une machine virtuelle, émulant le comportement d'un processeur.

Ces instructions de contrôle du certificat exécutable 4 en langage machine sont des structures binaires préalablement déterminées avant que celles ici ne soient transmises à l'application logicielle à vérifier.

Les instructions de contrôle du certificat exécutable 4 sont choisies de façon à ce que seule une application logicielle authentique puisse les exécuter pour produire un résultat identique à celui qui est attendu. Comme on le verra plus en détail ci-après, l'absence de dysfonctionnement de l'application logicielle authentique est obtenu en choisissant une suite d'instructions de contrôle de telle sorte que l'état du contexte mémoire d'une l'application logicielle à vérifier 1 après l'exécution de la suite d'instructions de contrôle soit identique à l'état du contexte mémoire de l'application logicielle avant l'exécution de la suite d'instructions de contrôle.

Le certificat exécutable 4 peut aussi être inséré dans le flot de données que l'application 1 est sensée traiter.

L'insertion de certificats exécutables dans un flot de données peut correspondre au cas où il est nécessaire d'authentifier une application de traitement de flux multimédia protégé, accessible à l'utilisateur à la condition que ce dernier se soit acquitté des obligations telles que définies par le vendeur des contenus. La source du flux multimédia peut être un point d'émission d'un réseau de diffusion, la mémoire persistante du terminal hôte, ou encore une unité de mémoire extractible du terminal hôte.

Les instructions de contrôle du certificat exécutable 4 sont choisies pour calculer 6 une signature 8 de l'application à contrôler 1, en utilisant 9 le contexte mémoire de l'application à contrôler 1, en cours d'exécution. L'utilisation du contexte mémoire par les instructions de contrôle est réalisé en allant chercher les valeurs de certains symboles (variables, fonctions,

instructions exécutable...) de l'application à vérifier en cours d'exécution. La récupération de ces valeurs dépend entre autre du modèle mémoire implémenté dans le processeur du terminal hôte.

En pratique, la suite d'instructions de contrôle du certificat exécutable 4 produit 6 une signature 8 qui dépend 9 du contexte mémoire de l'application logicielle à vérifier 1 et utilise cette signature pour lever la protection 10 des données protégées 2. Si l'application logicielle est authentique, les données 2 sont rendues accessibles et leur traitement 3 par l'application logicielle à vérifier 1 produira un résultat identique à celui d'une application authentique.

Dans le cas où le certificat exécutable 4 contenant les instructions de contrôle est inséré dans un flot de données, il est nécessaire de forcer l'application à exécuter ces instructions. Les instructions de contrôle sont alors programmées pour déchiffrer une partie du flot de données que l'application à vérifier doit traiter. Cela nécessite un traitement préalable du flot de données par chiffrement avant que ce flot ne soit accédé, pour traitement, par l'application à vérifier. L'algorithme de déchiffrement peut être implémenté dans les instructions de contrôle ou être disponible sous forme de fonction implémentée dans le terminal, et appelée par les instructions de contrôle. Les clés, si utilisées par l'algorithme de déchiffrement, sont calculées par les instructions de contrôle en utilisant des valeurs du contexte mémoire préalablement définies de l'application en cours d'exécution.

En référence à la figure 2, l'application logicielle à vérifier 11 interagit 12 avec un circuit sécurisé 13, de type carte à puce ou analogue.

La carte à puce 13 transmet 19 un certificat exécutable 15 contenant des instructions de contrôle qui sont chargées et exécutées 16 par l'application logicielle à vérifier 11. Ainsi, pour une application 11 nécessitant 12 une carte à puce 13 pour fonctionner, les instructions de contrôle 15 sont stockées dans la carte à puce 13 et envoyées à l'application à contrôler 11 par le biais du lien interactif 12.

L'application à contrôler 11 utilise 21 pour fonctionner des données 20 qu'elle récupère par interaction 12 avec la carte à puce 13. Ces données 20 contiennent 22 le certificat exécutable 15.

Les instructions de contrôle du certificat exécutable 15, lorsqu'elles sont stockées sur la carte à puce 13, sont chargées par l'application à contrôler 11 de façon à ce que celle ci les exécutent 16 selon le principe exposé en référence à la figure 1.

Une signature 18 de l'application logicielle à vérifier est produite 17 par les instructions de contrôle du certificat exécutable en utilisant 14 le contexte mémoire de l'application logicielle à contrôler.

Les instructions de contrôle du certificat exécutable interagissent 19 avec le circuit sécurisé 13 de façon à ce que la signature 18 de l'application contrôlée 11 soit transmise à une autre application de vérification 24 hébergée sur la carte à puce 13, considéré ici comme un environnement sécurisé. L'application de vérification 24 dans la carte à puce maintient pour chaque type de processeur et pour chaque application à contrôler, une table de correspondance entre les instructions de contrôle exécutables 15 et les résultats attendus.

Cette table de correspondance permet de vérifier 23 la validité de la signature calculée par le certificat exécutable 15. Si le résultat de la vérification est négatif, l'application de vérification hébergée dans la carte à puce interagit 12 avec l'application à contrôler afin de modifier le fonctionnement de cette dernière. Si la vérification est positive, la carte à puce 13 produit les données 20 dont l'application logicielle 11 a besoin pour fonctionner 21.

Dans un mode de réalisation, les certificats hébergés dans la carte à puce ou le circuit sécurisé changent selon une cadence ou condition choisie.

REVENDICATIONS

1. Procédé de vérification de l'intégrité d'une application logicielle exécutable dans un terminal hôte, caractérisé en ce qu'il comprend les étapes suivantes :

i) déterminer au moins une suite d'instructions de contrôle formant certificat exécutable (4,15) pour l'application logicielle, exécutable par ledit terminal hôte au cours de l'exécution de l'application logicielle à vérifier (1,11),

ii) sur le terminal hôte, exécuter l'application logicielle à vérifier (1,11), recevoir le certificat exécutable (4,15) ainsi déterminé lors de l'étape i), et exécuter la suite d'instructions de contrôle dudit certificat exécutable dans le contexte mémoire dudit terminal hôte,

iii) comparer le résultat ainsi obtenu par l'exécution des instructions de contrôle avec le résultat attendu d'une application logicielle authentique et,

iv) en cas de comparaison positive, continuer le cours de l'exécution de l'application logicielle à vérifier (1,11).

2. Procédé selon la revendication 1, dans lequel le terminal hôte est équipé d'un processeur caractérisé en ce que la suite d'instructions de contrôle formant certificat (4, 15) est codée en langage interprétable par ledit processeur du terminal hôte.

3. Procédé selon la revendication 1, dans lequel le terminal hôte est équipé d'une machine virtuelle apte à émuler un processeur, caractérisé en ce que la suite d'instructions de contrôle formant certificat (4, 15) est codée en langage interprétable par la machine virtuelle du terminal hôte.

4. Procédé selon l'une des revendications 1 à 3, caractérisé en ce que dans l'étape i) il est prévu d'établir, dans un environnement sécurisé, une carte du contexte mémoire de l'application logicielle authentique en cours d'exécution, et de déterminer, à partir des valeurs de cette carte mémoire, la suite d'instructions de contrôle destinée à former le certificat exécutable (4,15).

5. Procédé selon l'une des revendications 1 à 4, caractérisé en ce que dans l'étape ii), le certificat exécutable (4, 15) à destination du terminal hôte émane d'un circuit électronique de traitement physiquement séparé du terminal hôte.

6. Procédé selon l'une des revendications 1 à 5, caractérisé en ce que dans l'étape ii) la récupération des valeurs du contexte mémoire d'exécution se fait par lecture des valeurs aux adresses des différentes zones de la mémoire du terminal hôte, ces zones contenant les instructions exécutables et les données intrinsèques à l'application à vérifier.

7. Procédé selon l'une des revendications 1 à 6, caractérisé en ce que dans l'étape iii), le résultat obtenu par l'exécution de ladite suite d'instructions de contrôle (4,15) produit une signature de l'application à vérifier, cette signature étant calculée par ladite suite d'instructions de contrôle (4, 15) qui utilise les valeurs du contexte mémoire de l'application logicielle à vérifier en cours d'exécution de l'application.

8. Procédé selon l'une des revendications précédentes, caractérisé en ce que l'application logicielle comprend des instructions permettant de charger et d'exécuter dans sa carte de contexte mémoire ladite suite d'instructions de contrôle (4, 15) en substituant au moins une adresse d'exécution d'une instruction de ladite application logicielle par au moins une adresse d'instruction de la suite d'instructions formant certificat.

9. Procédé selon l'une des revendications précédentes, caractérisé en ce que la suite d'instructions de contrôle (4, 15) est choisie de telle sorte que l'état du contexte mémoire d'une l'application logicielle après l'exécution de la suite d'instructions de contrôle est identique et/ou sans modification de l'état du contexte mémoire de l'application logicielle avant l'exécution de la suite d'instructions de contrôle.

10. Procédé selon l'une quelconque des revendications 1 à 9, caractérisé en ce que la suite d'instructions formant certificat (4,15) est transportée dans un flux de données nécessaire à l'exécution de l'application logicielle à vérifier.

11. Procédé selon l'une quelconque des revendications 1 à 10, caractérisé en ce que l'application logicielle à vérifier est tout ou partie chiffrée, le déchiffrement correct de l'application logicielle étant réalisé en cas d'intégrité de l'application logicielle à vérifier.

12. Dispositif de vérification de l'intégrité d'une application logicielle destinée à être exécutée dans un terminal hôte pour la mise en œuvre du procédé selon l'une des revendications 1 à 11, caractérisé en ce qu'il comprend des moyens de traitement aptes à déterminer au moins une suite d'instructions de contrôle (4,15) pour l'application logicielle (1,11), exécutable par ledit terminal hôte au cours de l'exécution de l'application logicielle, et formant un certificat exécutable de ladite application logicielle, des moyens d'exécution pour exécuter la suite d'instructions formant certificat (4,15) sur le terminal hôte au cours de l'exécution de l'application logicielle, des moyens de comparaison pour comparer le résultat ainsi obtenu par l'exécution des instructions de contrôle avec le résultat attendu d'une application authentique, et des moyens aptes en cas de comparaison positive à continuer l'exécution de l'application logicielle à vérifier (1,11).

13. Dispositif selon la revendication 12, caractérisé en ce qu'il comprend une carte à puce ou tout autre circuit sécurisé apte à contenir la suite d'instructions de contrôle formant certificat (4,15), en ce que le terminal hôte est équipé d'un lecteur de carte à puce ou d'un moyen de communication avec le circuit sécurisé et en ce que les moyens d'exécution de l'application logicielle sont agencés pour aller chercher, dans la carte à puce ou le circuit sécurisé, la suite d'instructions formant certificat au cours de l'exécution de l'application logicielle à vérifier.

14. Dispositif selon la revendication 13, caractérisé en ce que le terminal hôte est apte à renvoyer à la carte à puce ou au circuit sécurisé la signature produite par la suite d'instructions de contrôle, et en ce que la carte à puce ou le circuit sécurisé comprend en outre une application logicielle de vérification apte à valider ou invalider l'authenticité de l'application logicielle à vérifier en fonction du résultat de la comparaison entre la signature produite par la suite d'instructions de contrôle et une valeur de la signature connue et préalablement stockée dans la carte à puce ou dans le circuit sécurisé.

15. Dispositif selon la revendication 14, caractérisé en ce qu'en cas de comparaison négative, la carte à puce est apte à modifier le fonctionnement de l'application logicielle à vérifier.

16. Dispositif selon la revendication 14 ou la revendication 15, caractérisé en ce qu'en cas de non transmission de la signature conformément à des conditions prédéterminées, la carte à puce est apte à modifier le fonctionnement de l'application logicielle à vérifier.

17. Dispositif selon l'une des revendications 12 à 16, caractérisé en ce qu'en cas de comparaison négative, le dispositif comprend en outre des moyens aptes à empêcher le fonctionnement de l'application logicielle dans le terminal hôte.

18. Dispositif selon l'une des revendications 12 à 17, caractérisé en ce que le terminal hôte appartient au groupe formé par les dispositifs de traitement des données, les décodeurs de télévision numérique, les équipements de visualisation de contenus multimédias, les micro-ordinateurs, les cartes à puces, les assistants personnels, les consoles de jeux, les téléphones mobiles ou analogues.

19. Dispositif selon l'une des revendications 12 à 18, caractérisé en ce que les moyens de traitement sont aptes à déterminer une pluralité de certificats

exécutables (4, 15), différents les un par rapport aux autres selon une cadence et/ou condition choisie.

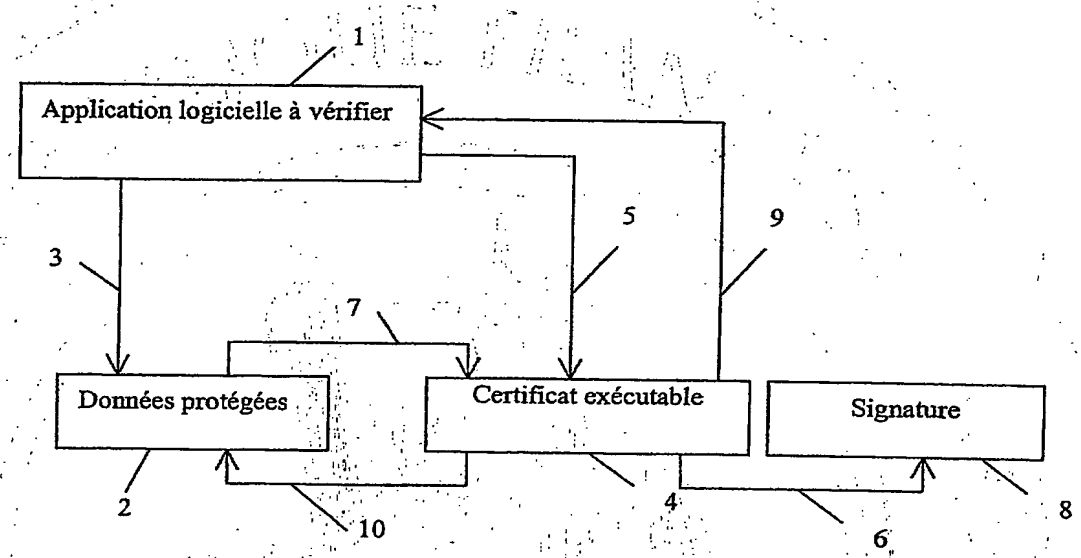
1
Application logicielle à vérifier
3
5
9
7
Données protégées
Certificat exécutable
Signature
2
10
4
6
8

Figure 1

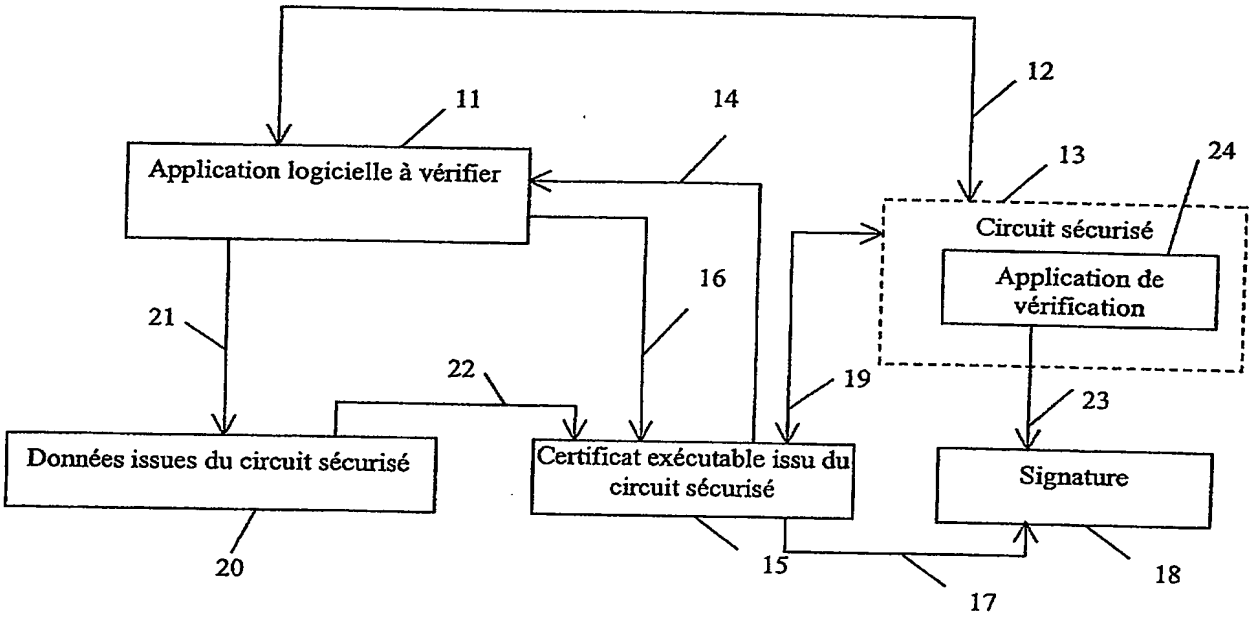
12
11
14
13
24
Application logicielle à vérifier
Circuit sécurisé
Application de vérification
16
21
22
19
23
Données issues du circuit sécurisé
Certificat exécutable issu du circuit sécurisé
Signature
20
15
17
18

Figure 2